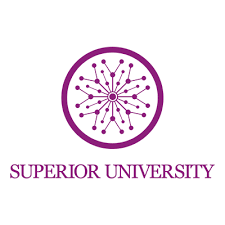
**TASK NO 9**

****

**Name:  
M. Zuhaib Anwar**

**Class :**

**SE – 3A**

**Roll No:**

**SU92-BSSEM-S24-005**

#include <iostream>

using namespace std;

class CNode {

public:

int value;

CNode\* next;

CNode(int val) : value(val), next(nullptr) {}

};

class CLL {

private:

CNode\* head;

public:

CLL() : head(nullptr) {}

void insertAtFirst(int val)

{

CNode\* newNode = new CNode(val);

if (!head)

{

head = newNode;

newNode->next = head;

} else {

CNode\* temp = head;

while (temp->next != head)

{

temp = temp->next;

}

temp->next = newNode;

newNode->next = head;

head = newNode;

}

}

void insert\_at\_last(int val)

{

CNode\* newNode = new CNode(val);

if (!head)

{

head = newNode;

newNode->next = head;

} else

{

CNode\* temp = head;

while (temp->next != head)

{

temp = temp->next;

}

temp->next = newNode;

newNode->next = head;

}

}

void insert\_at\_nth(int val, int n)

{

if (n < 1) return;

CNode\* newNode = new CNode(val);

if (n == 1)

{

insertAtFirst(val);

return;

}

CNode\* temp = head;

for (int i = 1; i < n - 1 && temp->next != head; i++)

{

temp = temp->next;

}

newNode->next = temp->next;

temp->next = newNode;

}

void insert\_at\_centre(int val)

{

if (!head) return;

CNode\* slow = head;

CNode\* fast = head;

while (fast->next != head && fast->next->next != head)

{

slow = slow->next;

fast = fast->next->next;

}

CNode\* newNode = new CNode(val);

newNode->next = slow->next;

slow->next = newNode;

}

void display()

{

if (!head)

{

cout << "The list is empty." << endl;

return;

}

CNode\* temp = head;

do

{

cout << temp->value << " ";

temp = temp->next;

} while (temp != head);

cout << endl;

}

void displayReverse()

{

cout << "Reverse display not supported in singly circular list directly." << endl;

}

};

int main() {

CLL list;

list.insertAtFirst(1);

list.insert\_at\_last(2);

list.insert\_at\_last(3);

list.insert\_at\_last(4);

list.insert\_at\_centre(5);

list.insert\_at\_nth(6, 3);

cout << "Circular List in Order: ";

list.display();

list.displayReverse();

}

OUTPUT:

